

**What is claimed is:**

1        1.    A method of initializing a computer system  
2 equipped with a debugging system, wherein the computer  
3 system has a CPU, a local, peripheral and expansion bus, a  
4 first and second bridge, and a ROM coupled to the expansion  
5 bus and storing a first BIOS code, and the debugging system  
6 is coupled to the peripheral bus, the method comprising the  
7 steps of:

8        routing data requests directed to the ROM to the local  
9        bus via the CPU;

10       transferring the data requests from the local bus to  
11       the peripheral bus via the first bridge;

12       switching the second bridge into a normal mode wherein  
13       the second bridge is enabled to respond to data

14       requests on the peripheral bus with the first

15       BIOS code in the ROM to be loaded in the CPU; and

16       switching the second bridge into a debugging mode

17       wherein the second bridge is disabled from

18       responding to data requests on the peripheral

19       bus, instead, the debugging system responds to

20       data requests with the second BIOS code to be

21       loaded in the CPU.

1       2.    The method as claimed in claim 1, wherein the  
2 second BIOS code is programmed by the debugging system.

1       3.    The method as claimed in claim 1, wherein the  
2 debugging system comprises:

3       an interface card coupled to the peripheral bus; and

4 a second computer system coupled to the interface card.

1 4. The method as claimed in claim 1 further  
2 comprising the step of:

3 when the second bridge is switched to debugging mode,  
4 retrieving and displaying contents of registers  
5 in the CPU via the debugging system.

1 5. The method as claimed in claim 1 further  
2 comprising the step of:

3 when the second bridge is switched to debugging mode,  
4 reading the first BIOS code from the ROM by the  
5 debugging system through the second bridge.

1 6. The method as claimed in claim 1 further  
2 comprising the step of:

3 when the second bridge is switched to debugging mode,  
4 overwriting the first BIOS code in the ROM with  
5 the second BIOS code by the debugging system  
6 through the second bridge.

1 7. The method as claimed in claim 1, wherein  
2 switching between the normal and debugging mode is performed  
3 by a strapping pin of the second bridge.

1 8. The method as claimed in claim 1, wherein the  
2 peripheral and expansion bus are a PCI and an ISA bus, and  
3 the first and second bridge are a north and south bridge,  
4 respectively.

1 9. The method as claimed in claim 1, wherein the  
2 second bridge responds the first data requests by sending a

Client Ref.: VIT02-0143

Our ref: 0608-8544-US/final/Vincent/Steve

3 device select signal to the peripheral bus, decoding  
4 addresses carried in the first data requests and retrieving  
5 the first BIOS code in the ROM corresponding to the  
6 addresses.

1 10. The method as claimed in claim 1, wherein the  
2 debugging system responds to the second data requests by  
3 sending a device select signal to the peripheral bus,  
4 decoding addresses carried in the second data requests and  
5 retrieving the second BIOS code therein corresponding to the  
6 addresses.

1 11. A computer system capable of being initialized by  
2 a debugging system, comprising:  
3 a CPU;  
4 a local, peripheral and expansion bus, wherein the CPU  
5 routes data requests to the local bus and the  
6 debugging system is coupled to the peripheral  
7 bus;  
8 a ROM coupled to the expansion bus and storing first  
9 BIOS code, wherein the data requests are directed  
10 to the ROM;  
11 a first bridge transferring the data requests from the  
12 local to the peripheral bus; and  
13 a second bridge switched between a normal mode wherein  
14 the second bridge is enabled to respond to data  
15 requests on the peripheral bus with the first  
16 BIOS code in the ROM to be loaded in the CPU and  
17 a debugging mode wherein the second bridge is  
18 disabled from responding to data requests on the

19            peripheral bus, instead, the debugging system  
20            responds to data requests with the second BIOS  
21            code to be loaded in the CPU.

1            12. The computer system as claimed in claim 11,  
2            wherein the second BIOS code is programmed by the debugging  
3            system.

1            13. The computer system as claimed in claim 12,  
2            wherein the debugging system comprises:  
3            an interface card coupled to the peripheral bus; and  
4            a second computer system coupled to the interface card.

1            14. The computer system as claimed in claim 11,  
2            wherein the debugging system retrieves and displays contents  
3            of registers in the CPU when the second bridge is switched  
4            to debugging mode.

1            15. The computer system as claimed in claim 11,  
2            wherein the debugging system reads the first BIOS code from  
3            the ROM through the second bridge when the second bridge is  
4            switched to debugging mode.

1            16. The computer system as claimed in claim 11,  
2            wherein the debugging system overwrites the first BIOS code  
3            in the ROM with the second BIOS code through the second  
4            bridge when the second bridge is switched to debugging mode.

1            17. The computer system as claimed in claim 11,  
2            wherein the second bridge has a strapping pin for switching  
3            between normal and debugging mode.

1        18. The computer system as claimed in claim 11,  
2 wherein the peripheral and expansion bus are a PCI and ISA  
3 bus, and the first and second bridge are a north and south  
4 bridge, respectively.

1        19. The computer system as claimed in claim 11,  
2 wherein the second bridge responds to data requests by  
3 sending a device select signal to the peripheral bus,  
4 decoding addresses carried in the data requests and  
5 retrieving the first BIOS code in the ROM corresponding to  
6 the addresses.

1        20. A peripheral/expansion bus bridge in a computer  
2 system capable of being initialized by a debugging system,  
3 wherein the computer system further comprises a CPU, a  
4 local, peripheral and expansion bus, a local/peripheral bus  
5 bridge, and a ROM coupled to the expansion bus to store a  
6 first BIOS code, and the debugging system is coupled to the  
7 peripheral bus, the peripheral/expansion bus bridge  
8 comprising:

9        a means for switching the peripheral/expansion bus  
10        bridge between a normal and debugging mode; and  
11        a decoder enabled to decode addresses carried by data  
12        requests of the CPU directed to the ROM, whereby  
13        the peripheral/expansion bus bridge sends a  
14        device select signal to the peripheral bus and  
15        retrieves the first BIOS code in the ROM  
16        corresponding to the addresses to be loaded in  
17        the CPU when the peripheral/expansion bus bridge  
18        is switched to normal mode, and disabled when the

Client Ref.: VIT02-0143

Our ref: 0608-8544-US/final/Vincent/Steve

19            peripheral/expansion bus bridge is switched to  
20            debugging mode;  
21        wherein, the debugging system instead responds to data  
22            requests with the second BIOS code to be loaded  
23            in the CPU when the peripheral/expansion bus  
24            bridge is switched to debugging mode.